PART4　変数の型と関数

目標

- 変数にはさまざまな型があることを理解する。
- 基本的な関数と算術演算子を使った計算をプログラミングできるようになる。

Chapter1　変数の型と関数

変数の型

　変数に記憶するデータが整数なのか，実数なのか，文字列なのかなどによって，コンピュータのメモリの中での記憶方法や必要なメモリのサイズが異なる。このようなデータの値の種類のことを〔①　　　　　　　〕とよぶ。Python では，最初に変数へ値を代入する時に変数の型が決定する。

知っておこう

Python 以外のプログラミング言語では，変数の型を明確に記述する（宣言する）必要があるものも多い。

変数の型	例
整数型 (int)	100
浮動小数点数型（実数型）(float)	3.14
文字列型 (str)	'tokyo'
論理型（bool)	True または False

```
#変数に値を代入
a = 100        # 整数型
b = 3.14       # 浮動小数点数型
c = 'tokyo'    # 文字列型 '（シングルクォート）で囲む
d = '03'       # 文字列型 '（シングルクォート）で囲む
e = True       # 論理型
```

ここが POINT

Python では = の記号は，等しいという意味ではなく，代入（変数に値を入れる）という意味になる。

ここが POINT

を付けるとコードの中にコメントと呼ばれるメモを入れることができる。

print関数

() の中の値を表示する。' ' で囲まれた文字列，計算した値，変数の値などを表示することができる。

```
# 変数に値を代入
a = 100
b = 3.14
c = 'tokyo'
d = '03'
e = True
# 変数の値を表示
print(a)
print(b)
print(c)
print(d)
print(e)
```

実行結果

```
100
3.14
tokyo
03
True
```

＋演算子

＋演算子は，演算子の前後が数値の場合は足し算を，文字列の場合は連結を行う。＋演算子の前後の変数の型が異なるとエラー (TypeError）となる。

```
a = 100
b = 3.14
c = 'tokyo'
d = '03'
print(a + b)
print(c + d)
```

実行結果

```
103.14
tokyo03
```

```
a = 100
b = 3.14
c = 'tokyo'
d = '03'
# エラーとなるケース
print(a + c)
print(b + d)
```

実行結果

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-1-da69fc8eb8fc> in <module>
      4 d = '03'
      5 # エラーとなるケース
----> 6 print(a + c)
      7 print(b + d)

TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

int 関数と str 関数

上記のエラーは，int 関数と str 関数を使うことで解消できる。

関数	説明
int	指定した数値または文字列を整数に変換
str	指定した数値を文字列に変換
float	指定した数値または文字列を浮動小数点数型に変換

```
a = 100
b = 3.14
c = 'tokyo'
d = '03'
print(str(a) + c)
print(b + int(d))
```

実行結果

```
100tokyo
6.140000000000001
```

ここが POINT

6.140000000000001 のように，浮動小数点数の計算では誤差（丸め誤差）が生じる。コンピュータの内部処理は 2 進法で行われており，10 進法の小数を正確に表現できないため，このような誤差が生じる。
このように，コンピュータは万能ではなく，処理には限界があることを知っておくことが大切である。

算術演算子

算術演算子	意味
a + b	足し算
a - b	引き算
a * b	掛け算
a / b	割り算
a // b	商
a % b	a を b で割った余り
a ** b	a の b 乗

```
a = 10
b = 3
print(a + b)  # 足し算
print(a - b)  # 引き算
print(a * b)  # 掛け算
print(a / b)  # 割り算
print(a // b) # a を b で割ったときの商
print(a % b)  # a を b で割ったときの余り
print(a ** b) # a の b 乗
```

実行結果

```
13
7
30
3.3333333333333335
3
1
1000
```

まとめ

- 変数には**型**があり，異なる型同士の連結や計算はできない。
- プログラミング言語には，あらかじめ用意された**関数**がある。

関数	説明
int	指定した数値または文字列を整数型に変換
str	指定した数値を文字列に変換
float	指定した数値または文字列を浮動小数点数型に変換
print	続く () 内の値を表示する

Chapter2　実践問題

問 1　Python で次の計算をするプログラムを作成しなさい。ただし，a = 9，b = 2 とする。

(1) a を b で割ったときの商

(2) a を b で割ったときの余り

(3) a の b 乗

問 2　a = 'shibuya', b = 109 のとき，「shibuya109」と表示するプログラムを Python で作成しなさい。

問 3　次の Python のプログラムを実行するとエラーになる。エラーが表示されないよう print(a + x + ' 歳です ') の部分を修正しなさい。

```
a = ' 私の年齢は '
x = 18
print(a + x + ' 歳です ')
```

問 4　次の Python のプログラムを実行するとエラーになる。エラーが出ないように print(x / y) の部分を修正しなさい。

```
x = '3.14'
y = 2
print(x / y)
```

＜MEMO＞

第6講　PART4　確認問題

問 1　次の Python のプログラムを実行すると表示される結果を，次の①～④のうちから選びなさい。

```
a = 3
b = 4
c = 5
print(a + b * c)
```

①　17　　②　23
③　35　　④　60

問 2　次の Python のプログラムを実行すると表示される結果を，次の①～④のうちから選びなさい。

```
a = 78
b = 8
print(a % b)
```

①　0　　②　3
③　6　　④　9

問 3　次の Python のプログラムを実行するとエラーとなる。このプログラムのエラーを解消する方法を，次の①～④のうちから選びなさい。

```
a = 'tokyo'
b = 100
print(a + b)
```

①　100 を '100' に変更する
②　100 を int(100) に変更する
③　print(a + b) を print(a - b) に変更する
④　'tokyo' を tokyo に変更する

＜MEMO＞

第６講　アルゴリズムとプログラミング

PART1　アルゴリズムの表現方法

問１　（ア）②　（イ）①　（ウ）③　（エ）④　（オ）⑤

問２　（ア）③　（イ）②　（ウ）①　（エ）⑤　（オ）④

問３　③

PART2　プログラミングの基本

問１　(1) 省略　(2) 省略　(3) 省略

PART3　ネットワークを利用したプログラミング

問１　(1) 省略　(2) 省略

PART4　変数の型と関数

問１　(1) `print(a // b)`

(2) `print(a % b)`

(3) `print(a ** b)`

問２

```
a = 'shibuya'
b = 109
print(a + str(b))
```

問３　`print(a + str(x) + '歳です')`

問４　`print(float(x) / y)`

PART5　モジュールとfor文・if文

問１　（ア）math　（イ）math.pi

問２　（ア）random　（イ）random.randint

問３　（ア）(10,15)　（イ）a == 10:

問４　（ア）range(4)　（イ）cnt == 0

PART6　配列と反復処理

問１　（ア）len(A)　（イ）A[i]

問２　（ア）temp　（イ）X[i]

問３　（ア）n　（イ）r

問４　（ア）Coin[i]　（イ）%

PART7　論理演算子と関数

問１　（ア）or　（イ）return

問２　(1)（ア）0　（イ）i

(2)（ウ）and　（エ）elif

問３　（ア）>　（イ）temp

PART8　【発展】Web APIや外部データの活用

問１　（ア）requests　（イ）'address1'

問２　（ア）pandas　（イ）matplotlib.pyplot

（ウ）'昨日までの観測史上１位の値（℃）'

第 6 講　PART4　確認問題　解答

問 1　②

解説：

「*」は掛け算，「+」は足し算を行う演算子である。したがって，「(a ＋ b *c)」は 3 ＋ 4 × 5 であるため，以下のようになる。

3 + 4 × 5 = 23

問 2　③

解説：

「%」は商の余りを求める演算子である。したがって,「(a % b)」は 78 を 8 で割った余りの 6 となる。

問 3　①

解説：

変数には型があり，異なる型どうしの連結はできない。そのため，変数 a と変数 b の型は揃える必要がある。変更前のプログラムでは，変数 a が文字列，変数 b は整数型である。選択肢①を実施すると，変数 b が文字列となるため，エラーは解消する。